****

**POO**

  NOME:   Nicole Fernanda

Prof. Dorival

Campinas

27/05/2025

**HERANÇA**

Se baseia em uma relação entre classes, em que uma classe herda atributos e métodos.

**MÃE**

**Superclasse**

**Subclasse**

**Filha**

Idade

brincar (brinquedo: str): None

endereço: str

sobrenome: str

comer: None

estudar: None

A classe que herda se chama subclasse (filha), e de onde os atributos e métodos são herdados é chamada de superclasse(mãe).

# Classe mãe - superclasse

class Mae:

def \_\_init\_\_(self, endereco, sobrenome):

self.endereco = endereco

self.sobrenome = sobrenome

def comer(self):

print("A mãe está comendo.")

# Classe filha - subclasse

class Filha(Mae):

def \_\_init\_\_(self, endereco, sobrenome, idade):

super().\_\_init\_\_(endereco, sobrenome)

self.idade = idade

def brincar(self, brinquedo):

print(f"A filha está brincando com {brinquedo}.")

menina = Filha("Rua das Flores, n° 123", "Silva", 8 )

# Métodos herdados

menina.comer()

menina.brincar("boneca")

# Atributos herdados

print("Elas moram na: ",menina.endereco)

print("O sobrenome das duas é:",menina.sobrenome)

print("Idade da filha: ",menina.idade)

**POLIMORFISMO**

**(**diferentes coisas podem ser tratadas da mesma forma**)**

Permite que um mesmo nome represente vários comportamentos diferentes.

class Animal:

def falar(self):

print("O animal faz um som")

class Cachorro(Animal):

def falar(self):

print("O cachorro late")

class Gato(Animal):

def falar(self):

print("O gato mia")

# Polimorfismo

animais = [Cachorro(), Gato()]

for animal in animais:

animal.falar()

**RELAÇÕES X DIFERENÇAS**

**RELAÇÕES**

A herança permite utilizar o mesmo nome de método, mas com ações diferentes. E o polimorfismo funciona por causa da herança, mas pode existir também com interfaces ou protocolos.

Então, o polimorfismo aproveita nomes de métodos com diferentes comportamentos, e a herança reutiliza atributos e métodos.

Ambos usam classes como base para estruturar o sistema.

**DIFERENÇAS**

Enquanto a herança faz relações entre classes, o polimorfismo é uma característica de métodos que se comportam de formas diferentes.

A herança permite que uma classe herde atributos de outra, e o polimorfismo permite métodos com o mesmo nome tenham comportamentos diferentes dependendo da classe.

A herança pode existir sem o polimorfismo, mas o polimorfismo precisa de herança ou interfaces para funcionar.

**VANTAGENS**

(Principal: reutilização do código)

A herança promove reutilização do código e criação de classes hierárquicas e oferece suporte ao polimorfismo, que torna o código flexível e de fácil manutenção.

**REFERÊNCIAS**

<https://www.alura.com.br/apostila-python-orientacao-a-objetos/heranca-e-classes-abstratas?srsltid=AfmBOor7f-UYsaKkIynJNtOtdFMvCqj8HesJ706Gaw6xjReXwmbh1Suz>

<https://awari.com.br/aprenda-sobre-heranca-e-polimorfismo-em-python-conceitos-essenciais-para-programadores-iniciantes-e-avancados/>

<https://www.dio.me/articles/heranca-e-polimorfismo-em-python-aprenda-a-estruturar-suas-hierarquias-de-classes>